(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 111 860 B1

(12) EUROPEAN PATENT SPECIFICATION

(45) Date of publication and mention of the grant of the patent: **09.11.2005 Bulletin 2005/45**

(51) Int Cl.7: **H04L 12/56**, H04L 12/46

(21) Application number: **00311286.9**

(22) Date of filing: **15.12.2000**

(54) Automatic protection switching using link-level redundancy supporting multi-protocol label switching

Automatischer Schutz von Vermittlungswegen mit Redundanz auf Verbindungsebene zur Unterstützung von MPLS

Commutation automatique de protection utilisant une redondance au niveau du lien supportant la commutation d'étiquettes multiprotocole

(84) Designated Contracting States:
DE FR GB

(30) Priority: **22.12.1999 US 471463**

(43) Date of publication of application:
27.06.2001 Bulletin 2001/26

(73) Proprietor: **Nortel Networks Limited**
St.Laurent, Quebec H4S 2A9 (CA)

(72) Inventors:
- **Fredette, Andre N.**
  Groton MA 01450 (US)
- **Andersson, Loa**
  125 33 Alvsjo (SE)
- **Doraswamy, Naganand**
  Arlington, MA 02474 (US)
- **Ghanwani, Anoop**
  North Billerica, MA 01862 (US)

(74) Representative: **Ertl, Nicholas Justin**
Elkington and Fife LLP,
Prospect House,
8 Pembroke Road
Sevenoaks, Kent TN13 1XR (GB)

(56) References cited:
US-A- 5 646 936 **US-A- 5 875 172**

- **VISWANATHAN A ET AL: "EVOLUTION OF MULTIPROTOCOL LABEL SWITCHING" IEEE COMMUNICATIONS MAGAZINE, IEEE SERVICE CENTER. PISCATAWAY, N.J, US, vol. 36, no. 5, 1 May 1998 (1998-05-01), pages 165-173, XP000752861 ISSN: 0163-6804**
- **LE FAUCHEUR F: "IETF Multiprotocol Label Switching (MPLS) Architecture" IEEE INTERNATIONAL CONFERENCE ON ATM, XX, XX, 22 June 1998 (1998-06-22), pages 6-15, XP002115225**

Printed by Jouve, 75001 PARIS (FR)

Description

[0001] The present invention relates to computer networks, and more specifically to a computer network that provides protection switching to reroute data packets in the event of a network link failure.

[0002] The various links of a computer network are paths between network nodes that communicate streams of data. In an Internet Protocol (IP) based computer network, data routing protocols such as Open Shortest Path First (OSPF), Intermediate System-Intermediate System (IS-IS), and Routing Information Protocol (RIP) are used to determine the path that data packets travel through the network. As a specific example, OSPF is a link-state protocol in the IP suite that enables routers to exchange information regarding topological changes within the network, as the link state database is modified. With this information, each router builds a shortest-path tree with itself as the tree root to identify the shortest path from itself to each destination and to build its routing table.

[0003] A router in a label switching network may sometimes explicitly route a particular data packet to another intermediate router that is not the ultimate destination of the packet, even though the two routers are not consecutive on the hop-by-hop path for that packet. For example, the affected data packet may be encapsulated inside a network layer packet whose destination is the intermediate router. This process establishes a "tunnel" between the two routers, and any data packet so handled is referred to as a "tunneled packet." A hop-by-hop tunnel follows the hop-by-hop path between the two routers. A tunneled packet that follows other than the hop-by-hop path is said to use an explicitly routed tunnel.

[0004] Occasionally, a link between two network routers may fail. When a link fails, the routing protocols are used to advertise the failure throughout the network. Most routers can detect a local link failure relatively quickly, but it takes the network as a whole a much longer time to converge. This convergence time is typically on the order of 10-60 seconds depending on the routing protocol and the size of the network. Eventually, all of the involved routers learn of the link failure and compute new routes for data packets to affected destinations. Once all the routers converge on a new set of routes, data packet forwarding proceeds normally.

[0005] While the network is converging after a link fails, transient loops can occur which consume valuable network bandwidth. A loop occurs when two or more intermediate routers in a given network path fail to update their internal representations of the network topology at the same time, and end up repeatedly passing data traffic between themselves rather than on to the correct destination. Loop prevention algorithms have been proposed to eliminate such transient loops. When using loop prevention algorithms, existing routes are maintained until the network has converged and the new routes have been proven to be loop-free. Loop prevention algorithms have the advantage that data packets flowing on unaffected routes are not disrupted while transient loops are eliminated. One drawback of loop prevention algorithms, however, is that data packets directed out of a failed link get lost, or "black holed," during the convergence. Moreover, since loop prevention algorithms also extend the convergence time somewhat while new routes are being verified to be loop-free, additional data may be lost on the failed link.

[0006] US 5875172 discloses a system for automatically restoring a transmission line in trouble in a communication network comprising a synchronous optical transmission network. A trouble working path is bypassed by provisioning a spare path. After restoration of the working path, path switching to the initial state is made.

[0007] US 5581543 discloses a network that is capable of responding to a failed link. When a failed link is detected, a "link failed" message is sent by a node that has detected the failed link to a route-determining node. The route determining node then re-calculates routes across the network so as to avoid the failed link, and sends the new routing information to other nodes. Once the other nodes have received the new routing information, the nodes implement the re-calculated routes so as to avoid routing data via the failed link. To avoid excessive loss of data to the failed link while the new routes are being calculated, an interim re-routing procedure is implemented at nodes adjacent to the failed link until the re-calculated routes can be implemented. The interim procedure diverts data at a node to a link located counter-clockwise to the failed link.

[0008] EP 0836344 discloses an asynchronous transfer mode (ATM) virtual path switching node. In one aspect, the node comprises a link monitoring unit for monitoring a fault in each of the plurality of links.

[0009] Viswanathan, A et al: "Evolution of Multiprotocol Label Switching" IEEE Communications Magazine, IEEE Service Center, Piscataway, N.J, US, vol. 36, no. 5, 1 May 1998, pages 165-173 provides an overview of Multiprotocol Label Switching architecture and design.

[0010] According to an aspect of the invention, there is provided a backup controller for providing protection switching in the event of a link failure of a routing node that delivers data packets to a computer network via a plurality of links, the backup controller comprising: means for pre-identifying, for at least one link of the routing node, a backup routing path for

forwarding affected data packets in the event of a failure of the at least one link; means for monitoring the plurality of links to determine when a link fails; means for attaching, when a link which has a pre-identified backup routing path fails, backup routing path instructions to affected data packets routed over the failed link; and means for forwarding the affected data packets via the backup routing path, wherein the backup routing path comprises an explicitly routed label switched path.

[0011] The invention also provides a data router and a computer network each comprising the backup controller described above.

[0012] According to another aspect of the invention, there is provided a method of providing protection switching in the event of a link failure of a computer network routing node that delivers data packets to a computer network via a plurality of links, the method comprising: pre-identifying, for at least one link of the routing node, a backup routing path for forwarding affected data packets in the event of a failure of the at least one link; monitoring the plurality of links to determine when a link fails; when a link which has a pre-identified backup routing path fails, attaching backup routing path instructions to affected data packets routed over the failed link; and forwarding the affected data packets via the backup routing path, wherein the backup routing path comprises an explicitly routed label switched path.

[0013] An embodiment of the invention provides a backup controller that provides protection switching in the event of a link failure of a routing node that delivers data packets to a computer network via a plurality of links. The computer network may use, for example, a label switching routing protocol. The backup controller has a backup path manager, a link monitor, and a backup packet processor. For at least one link of the routing node, the backup path manager identifies a backup routing path for forwarding affected data packets in the event of a failure of the at least one link. The link monitor monitors the plurality of links to determine when a link fails. When a link which has a backup routing path fails, the backup packet processor attaches backup routing path instructions to affected data packets routed over the failed link, and forwards the affected data packets via the backup routing path.

[0014] This backup controller may be used in a data router that delivers data packets to a computer network via a plurality of links. The data router provides protection switching in the event of a link failure. The data router also has a data interface for data packets to enter and exit the router, and a backup controller. Such a data router may also have a failed link recalculator that establishes a new network route to replace a failed link. The failed link recalculator may use a loop prevention algorithm, after a link failure, for determining that the network has converged and is loop-free.

[0015] The data router of the invention may also be included in a computer network having a plurality of data packet streams. The network has a plurality of subnetworks, each subnetwork having at least one application that generates a stream of data packets for transmission over the computer network; and a plurality of routers that deliver data packets to the network via a plurality of links, at least one router providing protection switching in the event of a link failure.

[0016] The invention also provides a method of providing protection switching in the event of a link failure of a computer network routing node that delivers data packets to a computer network via a plurality of links. The method includes identifying, for at least one link of the routing node, a backup routing path for forwarding affected data packets in the event of a failure of the at least one link; monitoring the plurality of links to determine when a link fails; when a link which has a backup routing path fails, attaching backup routing path instructions to affected data packets routed over the failed link; and forwarding the affected data packets via the backup routing path. In a further embodiment, a loop prevention algorithm may be used after a link failure to determine that the network has converged and is loop-free.

[0017] The method may be implemented by a computer program for use on a computer system.

[0018] The backup controller may further advertise a link failure to the network using a routing protocol. The backup routing path instructions may include a label stack based on Multi-Protocol Label Switching (MPLS), and the label stack may include labels for a packet source and a packet destination. The backup routing path may be a Label Switched Path (LSP), based on, for example, network topology information such as could be derived from a network protocol. Examples of the invention will now be described in detail with reference to the accompanying drawings, in which:

Fig. 1 is an illustration of a computer network which provides label switching-based backup path protection switching according to a representative embodiment.

Fig. 2 is an illustration of a network node router which supports backup paths according to a representative embodiment.

Fig. 3 is a flow chart illustration of the logical steps in a method of providing backup path protection switching according to a representative embodiment.

[0019] Representative embodiments of the present invention use a label switching protocol to establish backup paths with explicit routing for use in the event of a link failure in a computer network. A label is a short, fixed length, physically contiguous, locally significant identifier which is used to identify a given data stream in a label switching network. Multi-Protocol Label Switching (MPLS) is an example of a network layer-based label switching routing protocol that uses a forwarding paradigm based on label swapping to forward data traffic. Data forwarding between two adjacent network nodes using MPLS labels is known as a label switched hop. The concatenation of one or more label switched hops defines a Label Switched Path (LSP) that allows data packets to be forwarded from one MPLS node to another MPLS node by swapping labels. Explicit routing of an LSP is when the LSP is specified by the source of a data stream. The sequence of nodes defined by the LSP are defined by a layered stack of MPLS labels that typically may include a packet source label, a packet destination label, and labels for the nodes in the defined LSP.

[0020] In exemplary embodiments, each router establishes a backup path for each protected local link using MPLS-based Label Switched Path (LSP) tunnels. That is, a data packet sent over such a backup path follows an explicitly specified MPLS-LSP. Data packets are automatically rerouted on the backup link in the event that a protected link fails.

[0021] Fig. 1 is an illustration of an computer network capable of providing label switching-based protection switching in accordance with illustrative embodiments of the present invention. Network routers R1 **101** - R5 **105** are connected by primary network links **111-116.** Thus, in the network shown in Fig. 1, router devices R1 **101** and R2 **102** are connected by primary network link **111**, R1 **101** and R3 **103** are connected by primary network link **112**, R2 **102** and R3 **103** are connected by primary network link **113**, R2 **102** and R4 **104** are connected by primary network link **114**, R3 **103** and R5 **105** are connected by primary network link **115**, and R4 **104** and R4 **104** are connected by primary network link **116**.

[0022] For each primary network link, a backup path is established to be used in the event that the primary link fails. For example, if link L 3-5 **115** fails, router R3 **103** immediately starts to send data packets that would normally go to router R5 **105** over link L 3-5 **115** on LSP backup path L' 3-5 **121**. When router R5 **105** receives a packet via the backup path L' 3-5 **121**, it treats the packet just as if the packet had been received on the original failed link L 3-5 **115**. For clarity, Fig. 1 shows only one such backup path **121**, which represents the MPLS-based backup LSP for primary link **115** from router R3 **103** to router R5 **105**. In representative embodiments, there may be a backup path for every primary network link.

[0023] Fig. 2 is an illustration of a network router device which supports backup paths according to a representative embodiment. Fig. 3 is a flow chart of illustrative method steps in providing backup path protection with the router device of Fig. 2. Network node router **20** is a part of a computer network **22**, which are in mutual communication via a plurality of network node data links **21**. Router **20** also serves to connect one or more local area networks (LANs) **23** having one or more workstations 231. Data packets enter and exit the router **20** as controlled by a data interface driver **24** which is connected to the network node links **21**. Router **20** also includes a backup controller **25** having a link monitor **26**, a backup packet processor **27**, and a backup path manager **28**.

[0024] For each link to be protected, a backup path manager 28 identifies a backup path for forwarding affected data packets in the event that the protected link fails, step **301**. Backup paths can be hand configured or "automatically" computed using a link-state routing protocol, *e.g.*, Open Shortest Path First (OSPF). To automatically compute a backup path, the backup path manager **28** removes the primary link to be protected from its topology database, and then recomputes the shortest path to the destination router using a shortest-path algorithm. In typical embodiments, explicitly routed MPLS Label Switched Path (LSP) tunnels are used since the backup path follows a sub-optimal route that does not correspond to the normally routed path. Alternative embodiments may use another label switching protocol other than MPLS.

[0025] A link monitor **26** monitors protected links of the router for failure, step **302**. A link may fail, for example, if the path between two nodes is physically disrupted, or if a router loses power, disabling the connected links. Various different mechanisms may be used to detect such a link failure. For example, in a 100BASE-TX link integrity test, Fast Ethernet transceiver circuits continually monitor the receive data path for activity as a means of checking that the link is working correctly. The signaling system used for 100BASE-TX segments is based on the ANSI FDDI signaling system, which sends signals continually, even during idle periods of no network traffic. Therefore, activity on the receive data path is sufficient to provide a continual check of link integrity.

[0026] When the link monitor **26** initially determines that a protected link has failed, in step **303**, backup packet processor **27** attaches backup path instructions from the backup path manager **28**-for example, MPLS labels to affected data packets, which are forwarded through the network **22** over the backup for the failed link, step **304**. A predetermined period of time after the first detection of a link failure, the failure may be considered to be more than a temporary phenomenon, and the link may be considered to have positively failed. The router **20** then advertises the link failure to the network **22**, in step **305**, using a routing protocol, *e.g.*, Open Shortest Path First (OSPF).

[0027] New routes are determined to replace the failed link, step **306**. In step **307**, a diffusion-based loop-prevention algorithm determines when the network **22** has converged on new routes and is loop-free. To describe how loop prevention algorithms operate, it is important to first understand that most routing protocols use what are called "shortest-path" algorithms, which may be further sub-classified as being either distance-vector or link-state algorithms. A network node using a distance-vector algorithm, for example, knows the length of the shortest path from each neighboring node to every network destination. Based on this information, the node calculates the shortest path and next node in the path for each destination. Such nodes send to neighboring nodes update messages containing vectors of one or more entries each specifying the distance to a given destination. Receiving an update message may cause a node to generate an update message of its own. As a further example, a network node using a link-state algorithm (also called a topology broadcast algorithm) must know the topology of the entire network (or at least receive such information) in order to compute the shortest path to each network destination. Such nodes broadcast to every other node in the network, update messages containing the state of each of the node's adjacent links.

[0028] To avoid transient loops, loop prevention algorithms have been proposed based on diffusing computations, such as described by Garcia-Lunes-Aceves in *Loop-Free Routing Using Diffusing Computations*, IEEE/ACM Transactions on Networking, Vol. 1, No. 1, 1993. To that end, a family of distance vector algorithms are proposed which converge in a finite time after an arbitrary sequence of link cost or topological changes, being loop-free at any given instant, and having advantageous efficiency with respect to combined temporal, message, and storage complexities. Thus, loss of data packets is avoided by using the LSP tunnels to forward the affected data packets while the loop prevention algorithm is running.

[0029] Finally, in step **308**, once the new routes are confirmed to be loop-free, the routers revert from the back up path to the new routes, and the backup path manager **28** calculates new backup paths for the newly established routes.

[0030] Since representative embodiments use label switching, the present invention can operate successfully in any arbitrary network topology. It should be noted, however, that to realize full link-level protection, the network should have the property that for every two neighbors A and B connected by link L, there exists another path between A and B that does not include L. Various options may be employed with respect to network-level encapsulation on the original link. For example, the original network-layer encapsulation (*e.g.*, IP) may be tunneled in the backup LSP. If MPLS is used on the original link, then the labeled packet may be tunneled on the backup path using MPLS label stacking. Multiple independent link failures may be tolerated using multiple layers of tunneling.

[0031] Various embodiments of the invention, or portions thereof (*e.g.*, the link monitor **26**, the backup packet processor **27**, the backup path manager **28**, etc.), may be implemented in any conventional computer programming language. For example, representative embodiments may be implemented in a procedural programming language (*e.g.*, "C") or an object oriented programming language (*e.g.*, "C++" or "JAVA"). Alternative embodiments of the invention may be implemented as preprogrammed hardware elements (*e.g.*, application specific integrated circuits), or other related components.

[0032] Alternative embodiments of the invention may be implemented as a computer program product for use with a computer system. Such implementation may include a series of computer instructions fixed either on a tangible medium, such as a computer readable media (*e.g.*, a diskette, CD-ROM, ROM, or fixed disk), or transmittable to a computer system via a modem or other interface device, such as a communications adapter connected to a network over a medium. The medium may be either a tangible medium (*e.g.*, optical or analog communications lines) or a medium implemented with wireless techniques (*e.g.*, microwave, infrared or other transmission techniques). The series of computer instructions preferably embodies all or part of the functionality previously described herein with respect to the system. Those skilled in the art should appreciate that such computer instructions can be written in a number of programming languages for use with many computer architectures or operating systems. Furthermore, such instructions may be stored in any memory device, such as semiconductor, magnetic, optical or other memory devices, and may be transmitted using any communications technology, such as optical, infrared, microwave, or other transmission technologies. It is expected that such a computer program product may be distributed as a removable medium with accompanying printed or electronic documentation (*e.g.*, shrink wrapped software), preloaded with a computer system (*e.g.*, on system ROM or fixed disk), or distributed from a server or electronic bulletin board over the network (*e.g.*, the Internet or World Wide Web).

[0033] Although various exemplary embodiments of the invention have been disclosed, it should be apparent to those skilled in the art that various changes and modifications can be made without departing from the scope of the invention.

Claims

1. A backup controller (25) for providing protection switching in the event of a link failure of a routing node that delivers data packets to a computer network (22) via a plurality of links (21), the backup controller comprising:

   means for pre-identifying, for at least one link of the routing node, a backup routing path for forwarding affected data packets in the event of a failure of the at least one link;
   means for monitoring the plurality of links to determine when a link fails;
   means for attaching, when a link which has a pre-identified backup routing path fails, backup routing path instructions to affected data packets routed over the failed link; and
   means for forwarding the affected data packets via the backup routing path, wherein the backup routing path comprises an explicitly routed label switched path.

2. A backup controller according to claim 1, wherein

   the means for identifying comprises a backup path manager (28),
   the means for monitoring comprises a link monitor (26),
   the means for attaching and the means for for-

warding comprise a backup packet processor (27).

3. A backup controller according to claim 1 or 2, wherein the backup controller further comprises means for advertising a link failure to the network using a routing protocol.

4. A backup controller according to any preceding claim, wherein the backup routing path instructions include a label stack based on Multi-Protocol Label Switching.

5. A backup controller according to claim 4, wherein the label stack includes labels for a packet source and a packet destination.

6. A backup controller according to any preceding claim, wherein the explicitly routed label switched path is based on network topology information.

7. A backup controller according to claim 6, wherein the network topology information is derived from a network protocol.

8. A backup controller according to claim 1 or 2, wherein the computer network uses a label switching routing protocol.

9. A data router (20) for delivering data packets to a computer network (22) via a plurality of links, the router providing protection switching in the event of a link failure, the router comprising:

a data interface (24) for data packets to enter and exit the router; and

a backup controller (25) according to any one of claims 1 to 10.

10. A data router according to claim 9, further comprising a failed link recalculator for establishing a new network route to replace a failed link.

11. A data router according to claim 10, wherein the failed link recalculator further uses a loop prevention algorithm after a link failure for determining that the network has converged and is loop-free.

12. A computer network (22) having a plurality of data packet streams, the network comprising:

a plurality of subnetworks, each subnetwork having at least one application that generates a stream of data packets for transmission over the computer network; and

a plurality of routers (20) for delivering data packets to the network via a plurality of links (21), at least one router providing protection switching in the event of a link failure, the at least one router including:

a plurality of data interfaces (24) for streams of data packets to enter and exit the at least one router; and

a backup controller (25) according to any one of claims 1 to 8.

13. A computer network according to claim 12, further comprising a failed link recalculator for establishing a new network route to replace a failed link.

14. A computer network according to claim 13, wherein the failed link recalculator further uses a loop prevention algorithm after a link failure for determining that the network has converged and is loop-free.

15. A method of providing protection switching in the event of a link failure of a computer network routing node that delivers data packets to a computer network (22) via a plurality of links (21), the method comprising:

pre-identifying, for at least one link of the routing node, a backup routing path for forwarding affected data packets in the event of a failure of the at least one link (301);

monitoring the plurality of links to determine when a link fails (302);

when a link which has a pre-identified backup routing path fails, attaching backup routing path instructions to affected data packets routed over the failed link(303); and

forwarding the affected data packets via the backup routing path (304), wherein the backup routing path comprises an explicitly routed label switched path.

16. A method according to claim 15, further comprising advertising a link failure to the network using a routing protocol (305).

17. A method according to claim 15 or 16, further comprising establishing a new network route to replace a failed link (306).

18. A method according to any one of claims 15 to 17, further comprising using a loop prevention algorithm after a link failure to determine that the network has converged and is loop-free (307).

19. A method according to any one of claims 15 to 18, wherein the backup routing path instructions include a label stack based on Multi-Protocol Label Switching.

20. A method according to claim 19, wherein the label stack includes labels for a packet source and a packet destination.

21. A method according to any one of claims 15 to 20, wherein the explicitly routed label switched path is based on network topology information.

22. A method according to claim 21, wherein the network topology information is derived from a network protocol.

23. A method according to any one of claims 15 to 22, wherein the computer network uses a label switching routing protocol.

24. A computer program comprising computer program code means for performing all of the steps of the method of any one of claims 15 to 23 when said program is run on a computer.

25. A computer program according to claim 24 embodied on a computer readable medium.

Patentansprüche

1. Reserve-Steuerung (25) zur Bereitstellung einer Ersatzumschaltung im Fall eines Verbindungsstrecken-Ausfalls eines Routenführungs-Knotens, der Datenpakete an ein Computernetzwerk (22) über eine Vielzahl von Verbindungsstrecken (21) liefert, wobei die Reserve-Steuerung Folgendes umfasst:

   Einrichtungen zur Vor-Identifikation, für zumindest eine Verbindungsstrecke des Routenführungs-Knotens, eines Reserve-Routenführungspfades zur Weiterleitung betroffener Datenpakete im Fall eines Ausfalls der zumindest einen Verbindungsstrecke;
   Einrichtungen zur Überwachung der Vielzahl von Verbindungsstrecken, um festzustellen, wann eine Verbindungsstrecke ausfällt;
   Einrichtungen zum Anbringen von Reserve-Routenführungspfad-Befehlen an betroffene Datenpakete, die über die ausgefallene Verbindungsstrecke gelenkt werden, wenn eine Verbindungsstrecke, die einen vor-identifizierten Reserve-Routenführungspfad hat, ausfällt; und
   Einrichtungen zur Weiterleitung der betroffenen Datenpakete über den Reserve-Routenführungspfad, wobei der Reserve-Routenführungspfad einen explizit routengeführten etikettvermittelten Pfad umfasst.

2. Reserve-Steuerung nach Anspruch 1, bei der

   die Einrichtung zur Identifikation eine Reservepfad-Verwaltung (28) umfasst,
   die Einrichtung zur Überwachung eine Verbindungsstrecken-Überwachung (26) umfasst,
   die Einrichtung zum Anbringen und die Einrichtung zum Weiterleiten einen Reserve-Paket-Prozessor (27) umfassen.

3. Reserve-Steuerung nach Anspruch 1 oder 2, bei der die Reserve-Steuerung weiterhin Einrichtungen zur Ankündigung eines Verbindungsstrecken-Ausfalls an das Netzwerk unter Verwendung eines Routenführungs-Protokolls umfasst.

4. Reserve-Steuerung nach einem der vorhergehenden Ansprüche, bei der die Reserve-Routenführungspfad-Befehle einen Etikettstapel auf der Grundlage der Multiprotokoll-Etikettvermittlung einschließen.

5. Reserve-Steuerung nach Anspruch 4, bei der der Etikettstapel Etiketten für eine Paket-Quelle und ein Paket-Ziel einschließt.

6. Reserve-Steuerung nach einem der vorhergehenden Ansprüche, bei der der explizit routengeführte etikettvermittelte Pfad auf Netzwerk-Topologie-Information beruht.

7. Reserve-Steuerung nach Anspruch 6, bei der die Netzwerk-Topologie-Information von einem Netzwerk-Protokoll abgeleitet ist.

8. Reserve-Steuerung nach Anspruch 1 oder 2, bei der das Computernetzwerk ein etikettvermitteltes Routenführungsprotokoll verwendet.

9. Daten-Router (20) zur Lieferung von Paketen an ein Computernetzwerk (22) über eine Vielzahl von Verbindungsstrecken, wobei der Router eine Ersatzumschaltung im Fall eines Verbindungsstrecken-Ausfalls ergibt, wobei der Router Folgendes umfasst:

   eine Datenschnittstelle (24) für Datenpakete zum Eintritt in den Router und zum Verlassen des Routers; und
   eine Reserve-Steuerung (25) nach einem der Ansprüche 1-10.

10. Daten-Router nach Anspruch 9, der weiterhin eine Streckenausfall-Neuberechnung zur Ausbildung einer neuen Netzwerk-Route zum Ersatz einer ausgefallenen Verbindungsstrecke umfasst.

11. Daten-Router nach Anspruch 10, bei dem die Verbindungsstreckenausfall-Neuberechnungseinrichtung weiterhin einen Schleifenvermeidungs-Algorithmus nach einem Verbindungsstrecken-Ausfall zur Feststellung verwendet, dass das Netzwerk

konvergiert hat und schleifenfrei ist.

12. Computernetzwerk (22) mit einer Vielzahl von Datenpaket-Strömen, wobei das Netzwerk Folgendes umfasst:

eine Vielzahl von Teil-Netzwerken, wobei jedes Teil-Netzwerk zumindest eine Anwendung aufweist, die einen Strom von Datenpaketen zur Übertragung über das Computernetzwerk erzeugt; und
eine Vielzahl von Routern (20) zur Lieferung von Datenpaketen an das Netzwerk über die Vielzahl von Verbindungsstrecken (21) wobei zumindest ein Router eine Ersatzumschaltung im Fall eines Verbindungsstrecken-Ausfalls ergibt, wobei der zumindest eine Router Folgendes einschließt:

eine Vielzahl von Datenschnittstellen (24) für Ströme von Datenpaketen für den Eintritt in den und den Austritt aus dem zumindest einen Router; und
eine Reserve-Steuerung (25) nach einem der Ansprüche 1-8.

13. Computernetzwerk nach Anspruch 12, das weiterhin eine Verbindungsstreckenausfall-Neuberechnungseinrichtung zur Ausbildung einer neuen Netzwerk-Route zum Ersatz einer ausgefallenen Verbindungsstrecke umfasst.

14. Computernetzwerk nach Anspruch 13, bei dem die Verbindungsstreckenausfall-Neuberechnungseinrichtung weiterhin einen Schleifenvermeidungs-Algorithmus nach einem Verbindungsstrecken-Ausfall zur Feststellung verwendet, das das Netzwerk konvergiert hat und schleifenfrei ist.

15. Verfahren zur Bereitstellung einer Ersatzumschaltung in dem Fall eines Verbindungsstrecken-Ausfalls eines Computernetzwerk-Routenführungs-Knotens, der Datenpakete an ein Computernetzwerk (22) über eine Vielzahl von Verbindungsstrecken (21) liefert, wobei das Verfahren Folgendes umfasst:

Vor-Identifizieren, für zumindest eine Verbindungsstrecke des Routenführungs-Knotens, eines Reserve-Routenführungspfades zur Weiterleitung betroffener Datenpakete im Fall eines Ausfalls der zumindest einen Verbindungsstrecke (301);
Überwachen der Vielzahl von Verbindungsstrecken, um festzustellen, wann eine Verbindungsstrecke ausfällt (302);
wenn eine Verbindungsstrecke, die einen vor-identifizierten Reserve-Routenführungspfad hat, ausfällt, Anbringen von Reserve-Routenführungspfad-Befehlen an betroffenen Datenpaketen, die über die ausgefallene Verbindungsstrecke gelenkt werden (303); und
Weiterleiten der betroffenen Datenpakete über den Reserve-Routenführungspfad (304), wobei der Reserve-Routenführungspfad einen explizit routengeführten etikettvermittelten Pfad umfasst.

16. Verfahren nach Anspruch 15, das weiterhin die Ankündigung eines Verbindungsstrecken-Ausfalls an das Netzwerk unter Verwendung eines Routenführungs-Protokolls umfasst (305).

17. Verfahren nach Anspruch 15 oder 16, das weiterhin das Ausbilden einer neuen Netzwerk-Route zum Ersatz einer ausgefallenen Verbindungsstrecke umfasst (306).

18. Verfahren nach einem der Ansprüche 15-17, das weiterhin die Verwendung eines Schleifenvermeidungs-Algorithmus nach einem Verbindungsstrecken-Ausfall zur Feststellung umfasst, dass das Netzwerk konvergiert hat und schleifenfrei ist (307).

19. Verfahren nach einem der Ansprüche 15-18, bei dem die Reserve-Routenführungspfad-Befehle einen Etikettstapel einschließen, der auf der Multiprotokoll-Etikettvermittlung beruht.

20. Verfahren nach Anspruch 19, bei dem der Etikettstapel Etiketten für eine Paket-Quelle und ein Paket-Ziel einschließt.

21. Verfahren nach einem der Ansprüche 15-20, bei dem der explizit routengeführte etikettvermittelte Pfad auf Netzwerktöpologie-Information beruht.

22. Verfahren nach Anspruch 21, bei dem die Netzwerktopologie-Information von einem Netzwerk-Protokoll abgeleitet wird.

23. Verfahren nach einem der Ansprüche 15-22, bei dem das Computernetzwerk ein etikettvermittelndes Routenführungs-Protokoll verwendet.

24. Computerprogramm, das Computerprogramm-Codeeinrichtungen zur Durchführung aller der Schritte des Verfahrens nach einem der Ansprüche 15-23 umfasst, wenn das Programm auf einem Computer abläuft.

25. Computerprogramm nach Anspruch 24, das auf einem Computer lesbaren Medium verwirklicht ist.

Revendications

1. Un contrôleur de protection (25) pour fournir une commutation de protection dans le cas d'une défaillance affectant une liaison d'un noeud de routage qui fournit des paquets de données à un réseau informatique (22) par l'intermédiaire d'une pluralité de liaisons (21), le contrôleur de protection comprenant :

    un moyen pour pré-identifier, pour au moins une liaison du noeud de routage, un chemin de routage de secours pour acheminer des paquets de données affectés dans le cas d'une défaillance de l'au moins une liaison;
    un moyen pour surveiller la pluralité de liaisons pour déterminer le moment auquel une liaison devient défectueuse;
    un moyen pour adjoindre, lorsqu'une liaison qui a un chemin de routage de secours pré-identifié devient défectueuse, des instructions de chemin de routage de secours à des paquets de données affectés qui sont routés sur la liaison défectueuse; et
    un moyen pour acheminer par le chemin de routage de secours les paquets de données affectés, le chemin de routage de secours comprenant un chemin à commutation par étiquettes avec routage explicite.

2. Un contrôleur de protection selon la revendication 1, dans lequel
    le moyen pour identifier comprend un gestionnaire de chemin de secours (28),
    le moyen pour surveiller comprend un moniteur de liaisons (26),
    le moyen pour adjoindre et le moyen pour acheminer comprennent un processeur de paquets de secours (27).

3. Un contrôleur de protection selon la revendication 1 ou 2, dans lequel le contrôleur de protection comprend en outre un moyen pour déclarer une défaillance de liaison au réseau en utilisant un protocole de routage.

4. Un contrôleur de protection selon l'une quelconque des revendications précédentes, dans lequel les instructions de chemin de routage de secours comprennent une pile d'étiquettes basée sur une Commutation par Etiquettes Multiprotocole.

5. Un contrôleur de protection selon la revendication 4, dans lequel la pile d'étiquettes comprend des étiquettes pour une source de paquets et une destination de paquets.

6. Un contrôleur de protection selon l'une quelconque des revendications précédentes, dans lequel le chemin à commutation par étiquettes avec routage explicite est basé sur une information de topologie de réseau.

7. Un contrôleur de protection selon la revendication 6, dans lequel l'information de topologie de réseau est dérivée à partir d'un protocole de réseau.

8. Un contrôleur de protection selon la revendication 1 ou 2, dans lequel le réseau informatique utilise un protocole de routage à commutation par étiquettes.

9. Un routeur de données (20) pour fournir des paquets de données à un réseau informatique (22) par l'intermédiaire d'une pluralité de liaisons, le routeur fournissant une commutation de protection dans le cas d'une défaillance affectant une liaison, le routeur comprenant :

    une interface de données (24) pour permettre à des paquets de données d'entrer dans le routeur et de sortir de celui-ci; et
    un contrôleur de protection (25) selon l'une quelconque des revendications 1 à 10.

10. Un routeur de données selon la revendication 9, comprenant en outre un moyen de reprise de calcul de liaison défectueuse pour établir une nouvelle route de réseau pour remplacer une liaison défectueuse.

11. Un routeur de données selon la revendication 10, dans lequel un moyen de reprise de calcul de liaison défectueuse utilise en outre un algorithme de prévention de boucle après l'apparition d'une défaillance de liaison, pour déterminer que le réseau a convergé et est dépourvu de boucle.

12. Un réseau informatique (22) ayant une pluralité de flux de paquets de données, le réseau comprenant :

    une pluralité de sous-réseaux, chaque sous-réseau ayant au moins une application qui génère un flux de paquets de données pour la transmission sur le réseau informatique; et
    une pluralité de routeurs (20) pour fournir des paquets de données au réseau par l'intermédiaire d'une pluralité de liaisons (21), au moins un routeur fournissant une commutation de protection dans le cas d'une défaillance affectant une liaison, l'au moins un routeur incluant :

        une pluralité d'interfaces de données (24) pour permettre à des flux de paquets de données d'entrer dans l'au moins un routeur et de sortir de celui-ci; et
        un contrôleur de protection (25) selon l'une

quelconque des revendications 1 à 8.

13. Un réseau informatique selon la revendication 12, comprenant en outre un moyen de reprise de calcul de liaison défectueuse pour établir une nouvelle route de réseau pour remplacer une liaison défectueuse.

14. Un réseau informatique selon la revendication 13, dans lequel le moyen de reprise de calcul de liaison défectueuse utilise en outre un algorithme de prévention de boucle après l'apparition d'une défaillance de liaison pour déterminer que le réseau a convergé et est dépourvu de boucle.

15. Un procédé pour assurer une commutation de protection dans le cas d'une défaillance affectant une liaison d'un noeud de routage de réseau informatique qui fournit des paquets de données à un réseau informatique (22) par l'intermédiaire d'une pluralité de liaisons (21), le procédé comprenant les étapes consistant à :

pré-identifier, pour au moins une liaison du noeud de routage, un chemin de routage de secours pour acheminer des paquets de données affectés, dans le cas d'une défaillance de l'au moins une liaison (301);

surveiller la pluralité de liaisons pour déterminer le moment auquel une liaison devient défectueuse (302);

lorsqu'une liaison qui a un chemin de routage de secours pré-identifié devient défectueuse, adjoindre des instructions de chemin de routage de secours à des paquets de données affectés qui sont routés sur la liaison défectueuse (303); et

acheminer par le chemin de routage de secours (304) les paquets de données affectés, le chemin de routage de secours comprenant un chemin à commutation par étiquettes avec routage explicite.

16. Un procédé selon la revendication 15, comprenant en outre la déclaration d'une défaillance de liaison au réseau, en utilisant un protocole de routage (305).

17. Un procédé selon la revendication 15 ou 16, comprenant en outre l'établissement d'une nouvelle route de réseau pour remplacer la liaison défectueuse (306).

18. Un procédé selon l'une quelconque des revendications 15 à 17, comprenant en outre l'utilisation d'un algorithme de prévention de boucle après l'apparition d'une défaillance de liaison, pour déterminer que le réseau a convergé et est dépourvu de boucle (307).

19. Un procédé selon l'une quelconque des revendications 15 à 18, dans lequel les instructions de chemin de routage de secours comprennent une pile d'étiquettes basée sur la Commutation par Etiquettes Multiprotocole.

20. Un procédé selon la revendication 19, dans lequel la pile d'étiquettes comprend des étiquettes pour une source de paquets et une destination de paquets.

21. Un procédé selon l'une quelconque des revendications 15 à 20, dans lequel le chemin à commutation par étiquettes avec routage explicite est basé sur une information de topologie de réseau.

22. Un procédé selon la revendication 21, dans lequel l'information de topologie de réseau est obtenue à partir d'un protocole de réseau.

23. Un procédé selon l'une quelconque des revendications 15 à 22, dans lequel le réseau informatique utilise un protocole de routage à commutation par étiquettes.

24. Un programme informatique comprenant du code de programme informatique pour accomplir toutes les étapes du procédé de l'une quelconque des revendications 15 à 23, lorsque ce programme est exécuté sur un ordinateur.

25. Un programme informatique selon la revendication 24, incorporé sur un support lisible par ordinateur.

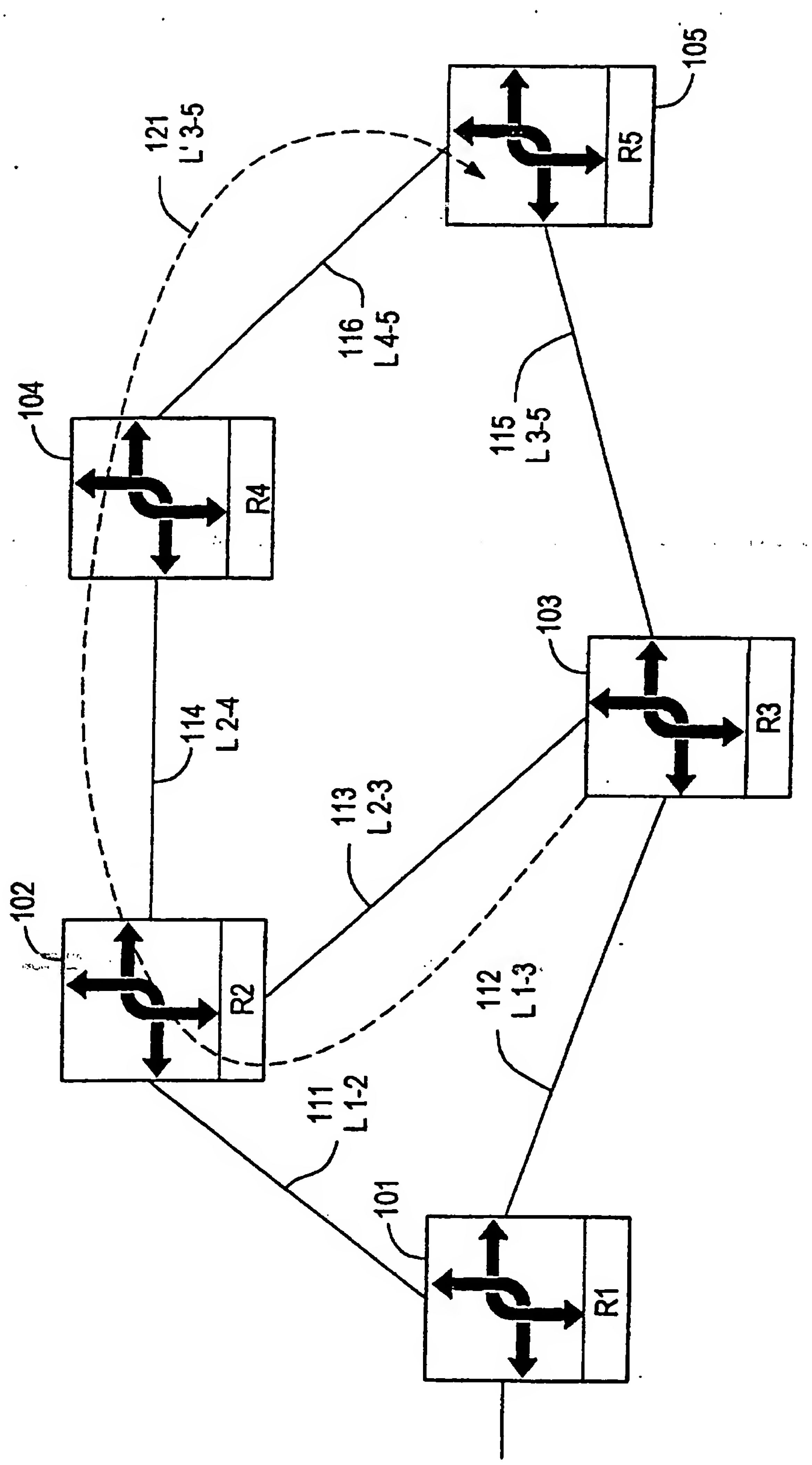
105
R5
121
L' 3-5
116
L 4-5
115
L 3-5
104
R4
103
R3
114
L 2-4
113
L 2-3
102
R2
112
L 1-3
111
L 1-2
101
R1

FIG. 1

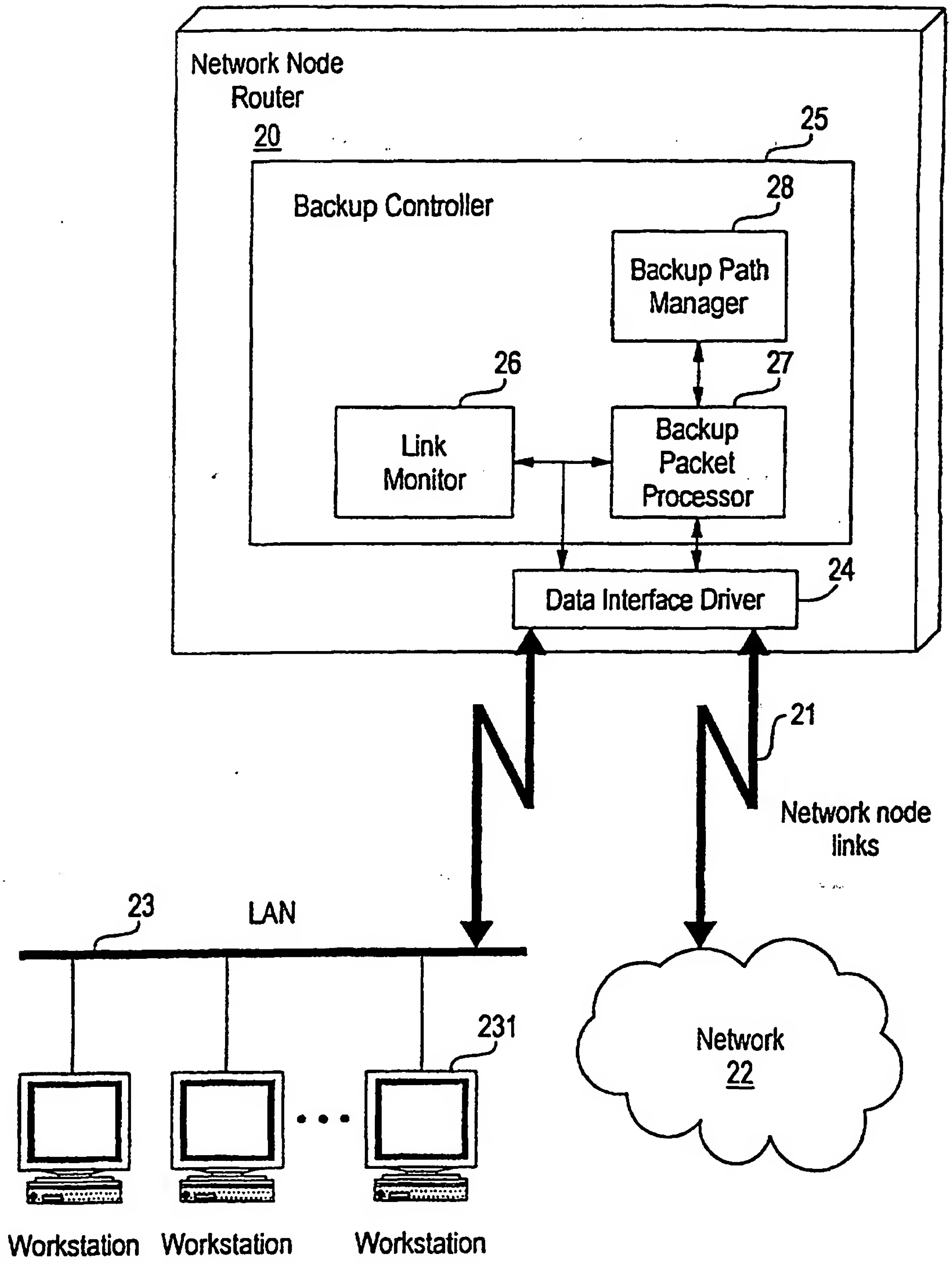
Network Node
Router
20
25
Backup Controller
28
Backup Path Manager
26
27
Link Monitor
Backup Packet Processor
Data Interface Driver
24
21
Network node links
23
LAN
231
Network
22
Workstation
Workstation
Workstation

FIG. 2

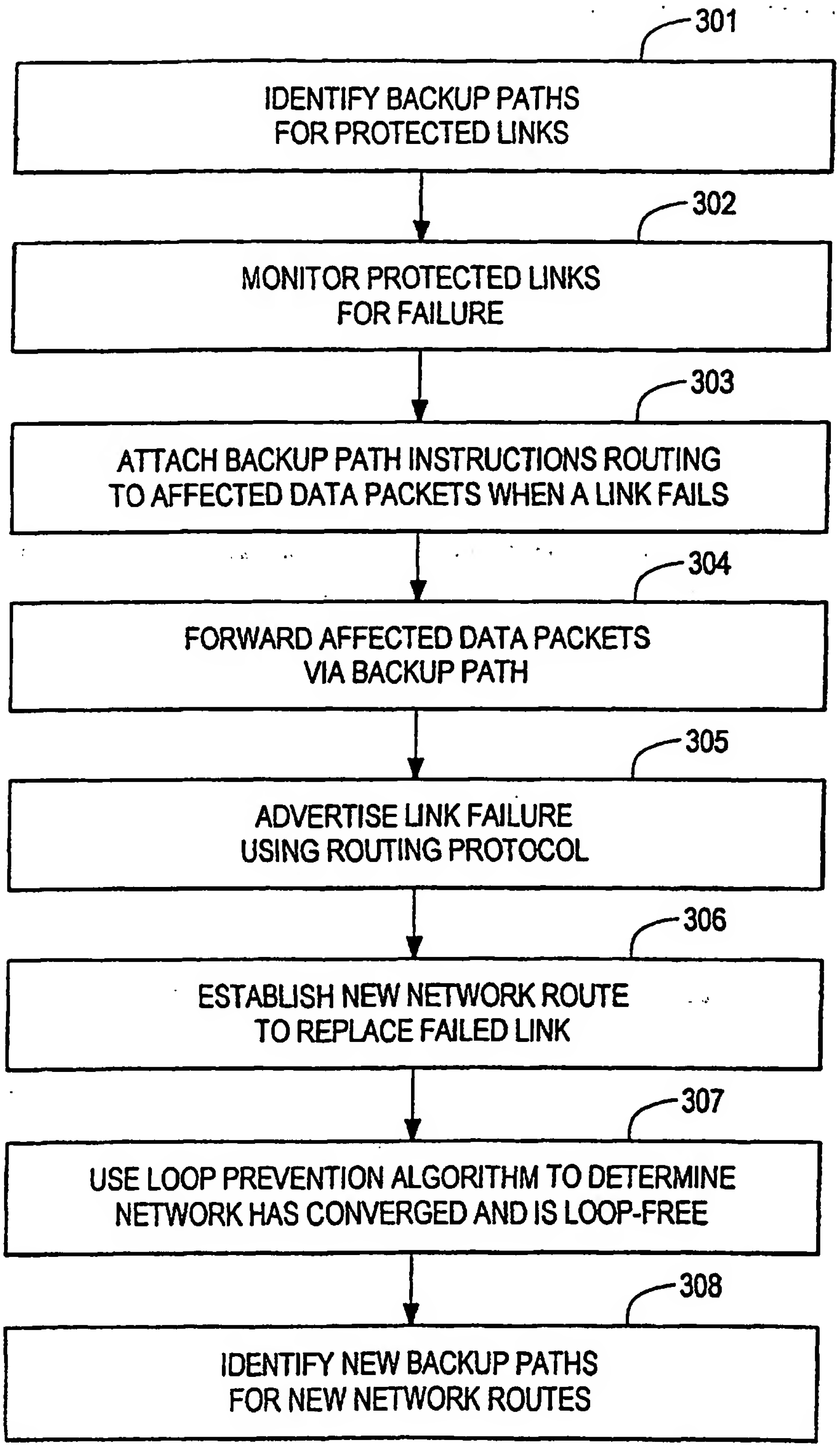
301
IDENTIFY BACKUP PATHS FOR PROTECTED LINKS
302
MONITOR PROTECTED LINKS FOR FAILURE
303
ATTACH BACKUP PATH INSTRUCTIONS ROUTING TO AFFECTED DATA PACKETS WHEN A LINK FAILS
304
FORWARD AFFECTED DATA PACKETS VIA BACKUP PATH
305
ADVERTISE LINK FAILURE USING ROUTING PROTOCOL
306
ESTABLISH NEW NETWORK ROUTE TO REPLACE FAILED LINK
307
USE LOOP PREVENTION ALGORITHM TO DETERMINE NETWORK HAS CONVERGED AND IS LOOP-FREE
308
IDENTIFY NEW BACKUP PATHS FOR NEW NETWORK ROUTES

FIG. 3